

Linux based 3G Multimedia Mobile-phone API Specification [Telephony Service] <Equipment Service>

Draft 2.2

NEC Corporation
Panasonic Mobile Communication Ltd.

Contents

4. EQUIPMENT SERVICE FUNCTIONS	1
4.1 START EQUIPMENT STATUS MONITORING	1
4.2 STOP EQUIPMENT STATUS MONITORING	5
4.3 SET EARPHONE MODE	6
4.4 GET EARPHONE CONNECTION STATUS	7
4.5 GET EARPHONE MODE	8
4.6 CHANGE MANNER MODE	9
4.7 GET MANNER MODE	10
4.8 START AND STOP VIBRATION	11
4.9 SET VIBRATION PATTERN	12
4.10 GET VIBRATION PATTERN NAME	13
4.11 GET VIBRATION PATTERN	14
4.12 GET CURRENT VIBRATION PATTERN	15
4.13 CHANGE SECRET MODE	16
4.14 GET SECRET MODE	17
4.15 CHANGE TERMINAL LOCK MODE	18
4.16 GET TERMINAL LOCK MODE	19
4.17 CHANGE PIM LOCK MODE	20
4.18 GET PIM LOCK MODE	21
4.19 CHANGE KEYPAD-ACTIVATED DIALING BAN MODE	22
4.20 GET KEYPAD-ACTIVATED DIALING BAN MODE	23
4.21 CHANGE CHARGE CONFIRMATION SOUND	24
4.22 GET BATTERY LEVEL	25
4.23 GET CHARGER STATUS	26
4.24 GET POWER ON STATUS	27
4.25 GET BATTERY PACK STATUS	28
4.26 GET CHARGE CONFIRMATION SOUND	29
4.27 NOTIFY FORCED POWER OFF	30
4.28 CHANGE SLEEP MODE OF REAR LCD	31
4.29 GET TERMINAL OPEN STATUS	32
4.30 GET BANNED-OPERATION STATUS	33
4.31 GET EQUIPMENT SERVICE MESSAGE	34
4.32 RESET USER SPECIFIED DATA	35
4.33 GET IMEI STATUS	36
4.34 GET USB CONNECTION STATUS	37
4.35 GET UIM INSERTION STATUS	38

4. EQUIPMENT SERVICE FUNCTIONS

4.1 Start equipment status monitoring

Classification	Equipment Service		
Function	Start equipment status monitoring		
Symbol	Elib_WDC_Request		
Syntax	int Elib_WDC_Request(ap_id, mask, func);		
Argument	Type	I/O	Description
ap_id	unsigned int	I	Application ID
Mask	int	I	Mask of the notification event Events with enums on are notified. WDCNotify_BattLv 1 : Battery level change notification WDCNotify_EarPhone 2 : Earphone connection status change notification WDCNotify_OpenClose 3 : Terminal open status change notification WDCNotify_Charge 4 : Charger status change notification WDCNotify_NvmAlarm 5 : Non-volatile alarm status change notification WDCNotify_Prohibit 8 : Banned-operation setup notification WDCNotify_ConInfo 9 : UIM connection information notice notification WDCNotify_UsbInf 12 : USB connection status change notification WDCNotify_IrInf 15 : Ir connection status change notification WDCNotify_Secret 13 : Secret status change notice notification WDCNotify_ALL 0 : All Equipment Service notifications
callback_func	MsbFunc	I	Callback function the event is notified
Return value	Type	I/O	Description
Ret	int	O	Processing result ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h msb/msb.h		
Functional description	<p>This function starts the monitoring the equipment status. The occurring of the event is notified to the application specified by ap_id. The events to be notified are described below.</p> <p>Events to be reported <u>Battery level change notification:</u> - Event notified an application from Equipment Service when Battery level changes.</p>		

Earphone connection status change notification:

- Event notified an application from Equipment Service when earphone connection status changes.

Terminal open status change notification:

- Event notified an application from Equipment Service when open or closed status changes.

Charger status change notification:

- Event notified an application from Equipment Service when charger status changes.

Non-volatile alarm status change notification :

- Event notified an application from Equipment Service when data preservation area breaks.

Banned-operation setup notification:

- The event of the initialization completion notified from the Equipment Service against the banned-operation setting of an application.

UIM connection information notice notification:

- Event notified an application from Equipment Service when UIM connection status changes.

USB connection status change notification:

- Event notified an application from Equipment Service when USB connection status changes.

Ir connection status change notification:

- Event notified an application from Equipment Service when Ir connection status changes.

Secret status change notice notification:

- The event of the status change notified from Equipment Service against the changing of secret mode of an application.

event structure(_ELIB_WDC_EVENT)

```
typedef struct {
    int    category;    The value is WDCNotify.
    int    subtype;     Same as the argument "mask".
    int    info;        Supplementary event information. See below
    int    subinfo;     The value is always zero.
    int    wdc_data;    Supplementary event information. See below
} _ELIB_WDC_EVENT ;
```

Subtype	Info
WDCNotify_BattLv	ELIB_WDC_BATTALM : Low-voltage alarm ELIB_WDC_BATTMIN : Lowest level ELIB_WDC_BATTLVL3 : Level 3 ELIB_WDC_BATTMAX : Highest level ELIB_WDC_BATTREC : Notice that the battery has recovered to normal voltage
WDCNotify_EarPhone	ELIB_WDC_EAR_ON : Earphone connection ON ELIB_WDC_EAR_OFF: Earphone connection OFF
WDCNotify_OpenClose	ELIB_WDC_OPEN : Terminal Open status ELIB_WDC_CLOSE : Terminal Closed status
WDCNotify_Charge	ELIB_WDC_CHARGE_OVER : External over-voltage detected ELIB_WDC_CHARGE_SET : "Charger connected" detected ELIB_WDC_CHARGE_NOTSET :

		"Charger not connected" detected ELIB_WDC_CHARGE_INF0 : Charger status "Charging" ELIB_WDC_CHARGE_INF1 : Charger status "Charge completed" ELIB_WDC_CHARGE_TEMP_ERR : Abnormal temperature detected ELIB_WDC_CHARGE_BATT_ERR : Battery anomaly detected ELIB_WDC_CHARGE_OUTBATT_START: Constant-voltage mode
	WDCNotify_NvmAlarm	ELIB_WDC_ALM_PA : Power alarm ELIB_WDC_ALM_PLL : PLL abnormal lock
	WDCNotify_Prohibit	ELIB_WDC_TRM_ON : Terminal lock setup ON ELIB_WDC_TRM_OFF : Terminal lock setup OFF ELIB_WDC_PIM_ON : PIM lock setup ON ELIB_WDC_PIM_OFF : PIM lock setup OFF ELIB_WDC_DIAL_ON : Keypad-activated dialing ban setup ON ELIB_WDC_DIAL_OFF : Keypad-activated dialing ban setup OFF
	WDCNotify_ConInfo	ELIB_WDC_CONNECT_NG : UIM anomaly (UIM failure) ELIB_WDC_CONNECT_OFF : UIM not connected (UIM not inserted) ELIB_WDC_CONNECT_ERR : UIM unusable (unallowable UIM inserted) * Those items other than ELIB_WDC_CONNECT_NG are notified of only when power is ON.
	WDCNotify_UsbInf	ELIB_WDC_USB_ON : USB connected ELIB_WDC_USB_OFF : USB not connected
	WDCNotify_IrInf	ELIB_WDC_Ir_ON : Ir connected ELIB_WDC_Ir_OFF : Ir not connected ELIB_WDC_Ir_BLINK : Ir blinking
	WDCNotify_Secret	ELIB_WDC_SECRET_MODE : Secret mode change ELIB_WDC_SECRET_DATA : Secret mode reference
	WDC_DATA of WDCNotify_Prohibit	

	31 10 9 8 7 6 5 4 3 2 1 0
	<p>Type : init</p> <p>Bits 7 to 31 ----- Unused</p> <p>Bit6 ----- Unused</p> <p>Bit5 ----- 1:Keypad-activated dialing ban setup ON 0:Keypad-activated dialing ban setup OFF</p> <p>Bit4 ----- Unused</p> <p>Bit3 ----- Unused</p> <p>Bit2 ----- 1:PIM lock setup ON 0:PIM lock setup OFF</p> <p>Bit1 ----- Unused</p> <p>Bit0 ----- 1:Terminal lock setup ON 0:Terminal lock setup OFF</p> <p>WDC_DATA of WDCNotify_Secret</p> <p>If the value of “info” is “ELIB_WDC_SECRECT_MODE”, the wdc_data is specified that the secret mode is Normal mode, Secret mode or Secret-only mode.</p> <p>If the value of “uinfo” is “EIIB_WDC_SECRECT_DATA”, the wdc_data is specified that the secret data is Secret data, or Normal data.</p>
Related message	-
Necessary procedure	
Note	
Prohibition	None
Use example	

4.2 Stop equipment status monitoring

Classification	Equipment Service		
Function	Stop equipment status monitoring		
Symbol	Elib_WDC_Cancel		
Syntax	int Elib_WDC_Cancel(ap_id, mask);		
Argument	Type	I/O	Description
ap_id	unsigned int	I	Application ID
Mask	int	I	Mask of the notification event to be deleted. Refer to "Start equipment status monitoring".
Return value	Type	I/O	Description
Ret	int	O	Processing result ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_NODATA : Not registered ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h		
Functional description	This function stops notifying of the event about specified equipment. For notification events, see "Start equipment status monitoring".		
Related message	-		
Necessary procedure	-		
Note			
Prohibition	None		
Use example			

4.3 Set earphone mode

Classification	Equipment Service		
Function	Set earphone mode		
Symbol	Elib_WDC_Set_EarphoneMode		
Syntax	int Elib_WDC_Set_EarphoneMode(mode);		
Argument	Type	I/O	Description
Mode	int	I	ELIB_WDC_EAR : Earphone only ELIB_WDC_SOUND : Earphone and speaker
Return value	Type	I/O	Description
Ret	int	O	Processing result ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h		
Functional description	<p>This function sets the earphone mode. With the argument "mode", it is set to (a) "earphone only" mode or (b) "earphone and speaker" mode. In "earphone only" mode, it sounds earphone, and in "earphone and speaker", it sounds both earphone and speaker.</p> <p>The completion of changing mode is notified by the event "Earphone connection status change notification".</p>		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.4 Get earphone connection status

Classification	Equipment Service		
Function	Get earphone connection status		
Symbol	Elib_WDC_Get_EarPhone		
Syntax	int Elib_WDC_Get_EarPhone();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
Ret	int	O	ELIB_WDC_EAR_ON : Earphone connected ELIB_WDC_EAR_OFF: Earphone not connected ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function gets the status whether earphone is connected or not.		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.5 Get earphone mode

Classification	Equipment Service		
Function	Get earphone mode		
Symbol	Elib_WDC_Get_EarPhoneMode		
Syntax	int Elib_WDC_Get_EarPhoneMode();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
Ret	int	O	ELIB_WDC_EAR : Earphone only ELIB_WDC_SOUND : Earphone and speaker ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function gets the current earphone mode.		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.6 Change manner mode

Classification	Equipment Service		
Function	Change manner mode		
Symbol	Elib_WDC_OnOff_Safety		
Syntax	int Elib_WDC_OnOff_Safety(mode);		
Argument	Type	I/O	Description
mode	int	I	ELIB_WDC_SAFETY_ON : The manner mode is ON. ELIB_WDC_SAFETY_OFF : The manner mode is OFF
Return value	Type	I/O	Description
ret	int	O	Processing result ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h		
Functional description	<p>This function changes the manner mode. When the manner mode is OFF, one of the following processing types is performed depending on the current communication status.</p> <ol style="list-style-type: none"> 1. in Waiting, The ringing tone is suppressed. 2. in Conversation, in three-party conversation, in progress, or in holding, The microphone amp gain is changed to high sensitive and manner mode start confirmatory sound is generated. 3. in Incoming Call Ringing tone is suppressed. 		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.7 Get manner mode

Classification	Equipment Service		
Function	Get manner mode		
Symbol	Elib_WDC_Get_Safety		
Syntax	int Elib_WDC_Get_Safety();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_SAFETY_ON : Manner mode ON ELIB_WDC_SAFETY_OFF : Manner mode OFF ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function gets the setting status of manner mode.		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.8 Start and stop vibration

Classification	Equipment Service		
Function	Start and stop vibration		
Symbol	Elib_WDC_StartStop_Vib		
Syntax	int Elib_WDC_StartStop_Vib(mode, VibID);		
Argument	Type	I/O	Description
mode	int	I	ELIB_WDC_VIB_DIRECT: Vibration starts with a pattern specified with VibID. ELIB_WDC_VIB_STOP : Vibration stops. ELIB_WDC_VIB_DIRECT_FOR_JAVA: i-appli software-dependent forced vibration
VibID	int	I	Specified a vibration ID. For the setting value, see "Get vibration pattern name". This argument is invalid only when the "mode" is ELIB_WDC_VIB_STOP.
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h		
Functional description	<p>The vibrator starts or stops vibrating. The vibrator should be stopped when i-appli software becomes inactive.</p> <p>This function is valid only for demonstrative vibration or i-appli software vibration. Other vibration starts and stops according to the incoming call process. It is not necessary to start and stop the vibration.</p>		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.9 Set vibration pattern

Classification	Equipment Service		
Function	Set vibration pattern		
Symbol	Elib_WDC_Set_VibMode		
Syntax	int Elib_WDC_Set_VibMode(VibID, kind);		
Argument	Type	I/O	Description
VibID	int	I	<p>A vibration pattern is specified.</p> <p>ELIB_WDC_VIB_PAT0 : Pattern No.0 (OFF)</p> <p>ELIB_WDC_VIB_PAT1 : Pattern No.1 (pattern 1)</p> <p>ELIB_WDC_VIB_PAT2 : Pattern No.2 (pattern 2)</p> <p>ELIB_WDC_VIB_PAT3 : Pattern No.3 (pattern 3)</p> <p>ELIB_WDC_VIB_PATMELO: Pattern No.4 (linked with a melody)</p> <p>The specified pattern number must be the one obtained with "Get vibration pattern name".</p>
kind	int	I	<p>ELIB_WDC_VIB_VOICE : Voice call ringing setup</p> <p>ELIB_WDC_VIB_MAIL : E-mail ringing setup</p> <p>ELIB_WDC_VIB_MSG_R : R-message ringing setup</p> <p>ELIB_WDC_VIB_MSG_F : F-message ringing setup</p> <p>ELIB_WDC_VIB_TV_VOICE : Videophone ringing setup</p>
Return value	Type	I/O	Description
ret	int	O	<p>Processing result</p> <p>ELIB_WDC_OK : Normal end</p> <p>ELIB_WDC_NG : Abnormal end</p> <p>ELIB_WDC_PARAERR : Parameter error</p>
Include file	srv_wdc.h		
Functional description	<p>This function sets the vibration pattern.</p> <p>When an incoming call type specified by "kind" causes, the vibrator operates with a vibration pattern specified by "VibID".</p>		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.10 Get vibration pattern name

Classification	Equipment Service				
Function	Get vibration pattern name				
Symbol	Elib_WDC_Get_VibTitle				
Syntax	int Elib_WDC_Get_VibTitle(No, Vib_DATA);				
Argument	Type	I/O	Description		
No	Int	I	Pattern No.		
Vib_DATA	_ELIB_WDC_VIB_TITLE *	O	Pointer to a structure for holding a vibration pattern name		
Return value	Type	I/O	Description		
ret	Int	O	ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_VIB_NODATA: Unregistered ELIB_WDC_PARAERR : Parameter error		
Include file	srv_wdc.h				
Functional description	This function gets the vibration pattern name and vibration ID. By calling this function with a pattern No. specified, an application can acquire the vibration pattern name and the corresponding vibration ID.				
	The pattern name and vibration ID are stored to the structure _ELIB_WDC_VIB_TITLE described below.				
	typedef struct { int VibID;				

4.11 Get vibration pattern

Classification	Equipment Service		
Function	Get vibration pattern		
Symbol	Elib_WDC_Get_VibMode		
Syntax	int Elib_WDC_Get_VibMode(kind);		
Argument	Type	I/O	Description
kind	int	I	ELIB_WDC_VIB_VOICE : Incoming voice call setup ELIB_WDC_VIB_MAIL : Incoming e-mail setup ELIB_WDC_VIB_MSG_R : Incoming R-message setup ELIB_WDC_VIB_MSG_F : Incoming F-message setup ELIB_WDC_VIB_TV_VOICE : Incoming videophone call setup * A parameter error is assumed if any other setup is made.
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_VIB_PAT0 : Pattern No. 0 (OFF) ELIB_WDC_VIB_PAT1 : Pattern No. 1 (pattern 1) ELIB_WDC_VIB_PAT2 : Pattern No. 2 (pattern 2) ELIB_WDC_VIB_PAT3 : Pattern No. 3 (pattern 3) ELIB_WDC_VIB_PATMELO: Pattern No. 4 (linked with a melody) ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h		
Functional description	This function gets the vibration pattern, specified by "ret", for the incoming call type, specified by "kind".		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.12 Get current vibration pattern

Classification	Equipment Service		
Function	Get current vibration pattern		
Symbol	Elib_WDC_Get_ActVib		
Syntax	int Elib_WDC_Get_ActVib(vib_pat);		
Argument	Type	I/O	Description
vib_pat	int *	O	<p>Pointer to an area for holding a vibration pattern. The return value indicates the current vibration pattern when vibration is in progress (valid only when vibration is in progress).</p> <p>ELIB_WDC_VIB_PAT1 : Vibration with pattern 1 in progress ELIB_WDC_VIB_PAT2 : Vibration with pattern 2 in progress ELIB_WDC_VIB_PAT3 : Vibration with pattern 3 in progress ELIB_WDC_VIB_PATMELO: Vibration linked with a melody in progress</p>
Return value	Type	I/O	Description
ret	int	O	<p>ELIB_WDC_VIB_OFF : Vibration not in progress ELIB_WDC_VIB_ON : Vibration in progress ELIB_WDC_NG : Abnormal end</p>
Include file	srv_wdc.h		
Functional description	This function reports the current vibrator operation status and sets to the area identified by "vib_pat".		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.13 Change secret mode

Classification	Equipment Service		
Function	Change secret mode		
Symbol	Elib_WDC_Set_SecretMode		
Syntax	int Elib_WDC_Set_SecretMode(mode);		
Argument	Type	I/O	Description
mode	int	I	Secret mode to be changed ELIB_WDC_NORMAL_MODE : Normal mode ELIB_WDC_SECRET_MODE : Secret mode ELIB_WDC_SECRET_SMODE: Secret-only mode
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h		
Functional description	This function changes a secret mode status of terminal. Secret mode has three modes of "Normal mode", "Secret mode", and "Secret-only mode". The completion of changing mode is notified by the event "Secret status change notice notification".		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.14 Get secret mode

Classification	Equipment Service		
Function	Get secret mode		
Symbol	Elib_WDC_Get_SecretMode		
Syntax	int Elib_WDC_Get_SecretMode();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_NORMAL_MODE : Normal mode ELIB_WDC_SECRET_MODE : Secret mode ELIB_WDC_SECRET_SMODE : Secret-only mode ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function gets a secret mode status of terminal.		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.15 Change terminal lock mode

Classification	Equipment Service		
Function	Change terminal lock mode		
Symbol	Elib_WDC_OnOff_TrmLock		
Syntax	int Elib_WDC_OnOff_TrmLock(mode);		
Argument	Type	I/O	Description
mode	int	I	Terminal lock mode to be changed ELIB_WDC_TRM_ON : Terminal lock (all-feature lock) ON ELIB_WDC_TRM_OFF : Terminal lock (all-feature lock) OFF
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h		
Functional description	This function changes the terminal lock (all-feature lock) mode. The mode is whether lock-on or lock-off. The completion of changing mode is notified by the event "Banned-operation setup notification".		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.16 Get terminal lock mode

Classification	Equipment Service		
Function	Get terminal lock mode		
Symbol	Elib_WDC_Get_TrmLock		
Syntax	int Elib_WDC_Get_TrmLock();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_TRM_ON : Terminal lock (all-feature lock) ON ELIB_WDC_TRM_OFF : Terminal lock (all-feature lock) OFF ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function reports the current terminal lock (all-feature lock) mode. The return value indicates whether the terminal lock (all-feature lock) is ON or OFF.		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.17 Change PIM lock mode

Classification	Equipment Service		
Function	Change PIM lock mode		
Symbol	Elib_WDC_OnOff_PIMLock		
Syntax	int Elib_WDC_OnOff_PIMLock(mode);		
Argument	Type	I/O	Description
mode	Int	I	PIM lock mode to be changed ELIB_WDC_PIM_ON : PIM lock ON ELIB_WDC_PIM_OFF : PIM lock OFF
Return value	Type	I/O	Description
Ret	Int	O	ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h		
Functional description	This function changes the PIM lock mode. The mode is whether lock-on or lock-off. The completion of changing mode is notified by the event "Banned-operation setup notification".		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.18 Get PIM lock mode

Classification	Equipment Service		
Function	Get PIM lock mode		
Symbol	Elib_WDC_Get_PIMLock		
Syntax	int Elib_WDC_Get_PIMLock();		
Arg	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
Ret	int	O	ELIB_WDC_PIM_ON : PIM lock ON ELIB_WDC_PIM_OFF : PIM lock OFF ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function reports the current PIM lock mode. The return value indicates whether the PIM lock is ON or OFF.		
Related message	None		
Necessary procedure			
Note			
Prohibition	None		
Use example			

4.19 Change keypad-activated dialing ban mode

Classification	Equipment Service		
Function	Change keypad-activated dialing ban mode		
Symbol	Elib_WDC_OnOff_DialSend		
Syntax	int Elib_WDC_OnOff_DialSend(mode);		
Argument	Type	I/O	Description
mode	int	I	Keypad-activated dialing ban mode to be changed ELIB_WDC_DIAL_ON : Keypad-activated dialing ban ON ELIB_WDC_DIAL_OFF : Keypad-activated dialing ban OFF
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h		
Functional description	This function changes Keypad-activated dialing ban mode. The mode is whether Keypad-activated dialing ban ON or OFF.		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.20 Get keypad-activated dialing ban mode

Classification	Equipment Service		
Function	Get keypad-activated dialing ban mode		
Symbol	Elib_WDC_Get_DialSend		
Syntax	int Elib_WDC_Get_DialSend();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_DIAL_ON : Keypad-activated dialing ban ON ELIB_WDC_DIAL_OFF : Keypad-activated dialing ban OFF ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function gets the keypad-activated dialing ban ON/OFF status. - The return value indicates whether the keypad-activated dialing ban is ON or OFF.		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.21 Change charge confirmation sound

Classification	Equipment Service		
Function	Change charge confirmation sound		
Symbol	Elib_WDC_Set_ChargeSound		
Syntax	int Elib_WDC_Set_ChargeSound(mode);		
Argument	Type	I/O	Description
mode	int	I	Confirmatory sound ON /OFF to be changed ELIB_WDC_CHARGESND_ON: Confirmatory sound ON ELIB_WDC_CHARGESND_OFF: Confirmatory sound OFF
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h		
Functional description	This function activates or deactivates confirmation sound at the start and end of battery charge.		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.22 Get battery level

Classification	Equipment Service		
Function	Get battery level		
Symbol	Elib_WDC_Get_BattLvl		
Syntax	int Elib_WDC_Get_BattLvl();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_BATTALM : Low-voltage alarm ELIB_WDC_BATTMIN : Lowest level ELIB_WDC_BATTLVL3 : Level 3 ELIB_WDC_BATTMAX : Highest level ELIB_WDC_BATTOVER: Overvoltage alarm ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function gets the current battery level status. - The return value indicates the current battery level. - ELIB_WDC_BATTMAX (highest level) is always returned when the battery is being charged.		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.23 Get Charger status

Classification	Equipment Service		
Function	Get Charger status		
Symbol	Elib_WDC_Get_Charge		
Syntax	int Elib_WDC_Get_Charge();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_CHARGE_OVER : External over-voltage detected ELIB_WDC_CHARGE_INF0 : Charger status "Charging" ELIB_WDC_CHARGE_INF1 : Charger status "Charge completed" ELIB_WDC_CHARGE_OFF : Charger not connected ELIB_WDC_CHARGE_TEMP_ERR : Abnormal temperature detected ELIB_WDC_CHARGE_BATT_ERR : Battery anomaly detected ELIB_WDC_CHARGE_OUTBATT_START: Constant-voltage mode ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function gets the charger status. - The return value indicates the current charger status.		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.24 Get power on status

Classification	Equipment Service		
Function	Get power on status		
Symbol	Elib_WDC_Get_PowerOnStat		
Syntax	int Elib_WDC_Get_PowerOnStat();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_PWRSW : Power SW-activated power ON /Battery coming off (allowable short power dip time elapsed) ELIB_WDC_PWRALM : Alarm-activated power ON ELIB_WDC_PWRWDT : Watchdog timer reset ELIB_WDC_PWRRST : Software reset ELIB_WDC_BATTOVER : Over-voltage alarm ELIB_WDC_HWRST : Hardware reset (recovery from short power dip) ELIB_WDC_PWRSET : Power source (battery or charger) attached ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function gets the cause of power ON. - The return value indicates the cause of power ON.		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.25 Get battery pack status

Classification	Equipment Service		
Function	Get battery pack status		
Symbol	Elib_WDC_Get_BattPack_Off		
Syntax	int Elib_WDC_Get_BattPack_Off();		
Argument	Type	I/O	Description
None	void	-	-
Retu	Type	I/O	Description
ret	int	O	ELIB_WDC_BATTOFF_YES : The battery pack was detached and attached again. ELIB_WDC_BATTOFF_NO : The battery pack is kept attached. ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function gets the information about whether the battery pack was detached and attached again. - The return value indicates whether the battery pack was detached and attached again.		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.26 Get charge confirmation sound

Classification	Equipment Service		
Function	Get charge confirmation sound		
Symbol	Elib_WDC_Get_ChargeSound		
Syntax	int Elib_WDC_Get_ChargeSound();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
Ret	int	O	ELIB_WDC_CHARGESND_ON : Set up ELIB_WDC_CHARGESND_OFF: Released ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function gets the information about whether to generate confirmatory sound at the start and completion of battery charge. - The return value indicates whether the confirmatory sound is set up.		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.27 Notify forced power off

Classification	Equipment Service		
Function	Notify forced power off		
Symbol	Elib_WDC_PowerOff		
Syntax	int Elib_WDC_PowerOff();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
Ret	int	O	ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function requires a forced power-OFF to MPP.		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.28 Change sleep mode of rear LCD

Classification	Equipment Service		
Function	Change sleep mode of rear LCD		
Symbol	Elib_WDC_OnOff_BackLcdSleep		
Syntax	int Elib_WDC_OnOff_BackLcdSleep (mode);		
Argument	Type	I/O	Description
mode	int	I	Rear LCD sleep mode to be changed ELIB_WDC_ON: Sleep mode ON ELIB_WDC_OFF: Sleep mode OFF
Return value	Type	I/O	Description
Ret	int	O	ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h		
Functional description	This function changes the rear LCD sleep mode. The mode is whether sleep mode ON or OFF.		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.29 Get terminal open status

Classification	Equipment Service		
Function	Get terminal open status		
Symbol	Elib_WDC_Get_OpenClose		
Syntax	int Elib_WDC_Get_OpenClose();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_OPEN : Open status ELIB_WDC_CLOSE : Closed status ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function gets the information about whether the mobile terminal is open or closed (if it is a folding type). - The return value indicates whether the mobile terminal is open or closed.		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.30 Get banned-operation status

Classification	Equipment Service															
Function	Get banned-operation status															
Symbol	Elib_WDC_Get_Prohibit_Status															
Syntax	int Elib_WDC_Get_Prohibit_Status(pro_sts);															
Argument	Type	I/O	Description													
pro_sts	int *	O	Banned-operation setup status See "Functional description."													
Return value	Type	I/O	Description													
ret	int	O	ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error													
Include file	srv_wdc.h															
Functional description	This function gets the currently banned operations setup status. - The following setup statuses are returned. 31 10 9 8 7 6 5 4 3 2 1 0 <table border="1"><tr><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr></table> Type : init															
Bits 7 to 31 ----- Unused Bit6 ----- Unused Bit5 ----- 1:Keypad-activated dialing ban setup ON 0:Keypad-activated dialing ban setup OFF Bit4 ----- Unused Bit3 ----- Unused Bit2 ----- 1:PIM lock setup ON 0:PIM lock setup OFF Bit1 ----- Unused Bit0 ----- 1:Terminal lock setup ON 0:Terminal lock setup OFF																
Related message																
Necessary procedure																
Note																
Prohibition																
Use example																

4.31 Get equipment service message

Classification	Equipment Service		
Function	Get equipment service message		
Symbol	Elib_WDC_Get_Message		
Syntax	int Elib_WDC_Get_Message(type, buflen, buff);		
Argument	Type	I/O	Description
type	Int	I	Message types ELIB_WDC_MSG_BATTALM Low-voltage alarm message ELIB_WDC_MSG_CHARGEOVER External over-voltage message ELIB_WDC_MSG_UIMNG UIM anomaly message ELIB_WDC_MSG_UIMOFF UIM not connected message ELIB_WDC_MSG_UIMERR UIM unusable message ELIB_WDC_MSG_UNCORR Wrong card inserted message
buflen	size_t	I	Size of the "buffer" for acquiring character strings * The return value is ELIB_WDC_NG if the specified buffer size is insufficient for holding all the character strings involved.
buff	unsigned char *	I/O	Pointer to a buffer which holds character strings * The buffer shall be secured by the APL that uses it.
Return value	Type	I/O	Description
ret	int	O	Normal end: The specified character string length is returned. Abnormal end: ELIB_WDC_NG Parameter error: ELIB_WDC_PARAERR
Include file	srv_wdc.h		
Functional description	This function stores the wording corresponding to the message type.		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.32 Reset user specified data

Classification	Equipment Service		
Function	Reset user specified data		
Symbol	Elib_WDC_UsrSetReset		
Syntax	int Elib_WDC_UsrSetReset(mode);		
Argument	Type	I/O	Description
mode	int	I	Specified reset mode ELIB_WDC_NORMAL_RESET: Normal reset
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h		
Functional description	This function resets each user-specified item (feature).		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.33 Get IMEI status

Classification	Equipment Service		
Function	Get IMEI status		
Symbol	Elib_WDC_Get_IMEI		
Syntax	int Elib_WDC_Get_IMEI(data);		
Argument	Type	I/O	Description
data	_ELIB_WDC_IMEI *	O	Pointer to a structure for holding IMEI information See "Functional description" for detailed descriptions about the structure.
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_OK : Normal end ELIB_WDC_NG : Abnormal end ELIB_WDC_PARAERR : Parameter error
Include file	srv_wdc.h		
Functional description	<p>This function gets IMEI information. The resulting ASCII character string is stored to the _ELIB_WDC_IMEI structure.</p> <p><_ELIB_WDC_IMEI information structure></p> <pre>typedef struct tagELIB_WDC_IMEI { unsigned char tac[6]; Type Approval Code unsigned char fac[2]; Final Assembly Code unsigned char snr[6]; Serial Number unsigned char spare[1]; Check Digit } _ELIB_WDC_IMEI;</pre>		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.34 Get USB connection status

Classification	Equipment Service		
Function	Get USB connection status		
Symbol	Elib_WDC_Get_USBInfo		
Syntax	int Elib_WDC_Get_USBInfo();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
ret	int	O	ELIB_WDC_USB_ON : USB connected ELIB_WDC_USB_OFF: USB not connected ELIB_WDC_NG : Abnormal end
Include file	srv_wdc.h		
Functional description	This function gets the USB connection status is referenced. - The return value indicates whether the USB is connected.		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			

4.35 Get UIM insertion status

Classification	Equipment Service		
Function	Get UIM insertion status		
Symbol	Elib_WDC_Get_UIMStatus		
Syntax	int Elib_WDC_Get_UIMStatus();		
Argument	Type	I/O	Description
None	void	-	-
Return value	Type	I/O	Description
ret	int	O	UIM card insertion status (See "Functional description."): ELIB_WDC_OK ELIB_WDC_CONNECT_NG ELIB_WDC_CONNECT_OFF ELIB_WDC_CONNECT_ERR ELIB_WDC_FAIL_CARD ELIB_WDC_NG
Include file	srv_wdc.h		
Functional description	<p>This function gets the UIM card insertion status.</p> <ul style="list-style-type: none"> - The return value indicates whether the UIM card has been inserted. <p><UIM card insertion status></p> <p>ELIB_WDC_OK : UIM connected normally (connected with no error message received)</p> <p>ELIB_WDC_CONNECT_NG: UIM anomaly (UIM failure occurred)</p> <p>ELIB_WDC_CONNECT_OFF: UIM not connected (no UIM inserted*)</p> <p>ELIB_WDC_CONNECT_ERR: UIM unusable</p> <p>ELIB_WDC_FAIL_CARD : Wrong UIM card inserted (there is a non-W-CDMA (3GPP standard) application in the UICC or there is no application at all in the UICC.*)</p> <p>ELIB_WDC_NG : Abnormal end</p> <p>* Set up at power-ON time.</p>		
Related message			
Necessary procedure			
Note			
Prohibition			
Use example			